



[Subscribe \(Full Service\)](#) [Register \(Limited Service, Free\)](#) [Login](#)

Search: The ACM Digital Library The Guide

object oriented and abstraction and machine independent



THE ACM DIGITAL LIBRARY

[Feedback](#) [Report a problem](#) [Satisfaction survey](#)

Terms used

object oriented and abstraction and machine independent

Found 101,734 of 158,639

Sort results by

relevance

[Save results to a Binder](#)

[Try an Advanced Search](#)

Display results

expanded form

[Search Tips](#)

[Try this search in The ACM Guide](#)

Open results in a new window

Results 1 - 20 of 200

Result page: [1](#) [2](#) [3](#) [4](#) [5](#) [6](#) [7](#) [8](#) [9](#) [10](#) [next](#)

Best 200 shown

Relevance scale

- 1 [A general, fine-grained, machine independent, object-oriented language](#)

Birger Andersen

May 1994 **ACM SIGPLAN Notices**, Volume 29 Issue 5

Full text available: [pdf\(653.50 KB\)](#) Additional Information: [full citation](#), [abstract](#), [index terms](#)

This paper introduces the general-purpose object-oriented programming language Ellie which supports machine independent fine-grained objects and parallelism. As something particular, classes, types, blocks, and methods are abstracted by first class objects/citizens called Ellie objects. Ellie demonstrates new approaches for abstraction and code reuse in parallel programming. The goals of Ellie have been to obtain an extremely flexible, machine independent, parallel language. Ellie tries to meet t ...

- 2 [Layered virtual machine/object-oriented design](#)

Ken Shumate

July 1988 **Proceedings of the fifth Washington Ada symposium on Ada**

Full text available: [pdf\(1.09 MB\)](#) Additional Information: [full citation](#), [references](#), [citations](#), [index terms](#)

- 3 [Concepts and paradigms of object-oriented programming](#)

Peter Wegner

August 1990 **ACM SIGPLAN OOPS Messenger**, Volume 1 Issue 1

Full text available: [pdf\(5.62 MB\)](#) Additional Information: [full citation](#), [abstract](#), [citations](#), [index terms](#)

We address the following questions for object-oriented programming: *What is it? What are its goals? What are its origins? What are its paradigms? What are its design alternatives? What are its models of concurrency? What are its formal computational models? What comes after object-oriented programming?* Starting from software engineering goals, we examine the origins and paradigms of object-oriented programming, explore its language design alternativ ...

- 4 [Experiences in applying the layered virtual machine/object-oriented development methodology to an Ada design effort](#)

C. Meyer, M. Wallis, M. Meier

January 1989 **Proceedings of the conference on Tri-Ada '89: Ada technology in context: application, development, and deployment**

Full text available:  pdf(810.63 KB) Additional Information: [full citation](#), [abstract](#), [references](#), [index terms](#)

When adopting a new design methodology for an Ada project, management must be aware of the various implications. Providing thorough training, establishing usage standards, integrating the methodology with an overall management plan, and assuring additional design guidance - if needed to supplement the design methodology - must be given due consideration early on in the project. Failure to do so will have a negative impact on the project.

5 [Virtual memory and backing storage management in multiprocessor operating systems using object-oriented design techniques](#) 

V. F. Russo, R. H. Campbell

September 1989 **ACM SIGPLAN Notices , Conference proceedings on Object-oriented programming systems, languages and applications**, Volume 24 Issue 10

Full text available:  pdf(1.19 MB) Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#)

The Choices operating system architecture [3, 4, 15] uses class hierarchies and object-oriented programming to facilitate the construction of customized operating systems for shared memory and networked multiprocessors. The software is being used in the Tapestry Parallel Computing Laboratory at the University of Illinois to study the performance of algorithms, mechanisms, and policies for parallel systems. This paper describes the architectural design and class hierarchy of ...

6 [Technical correspondence: The reflective nitrO abstract machine](#) 

Francisco Ortin, Juan Manuel Cueva, Ana Belen Martinez

June 2003 **ACM SIGPLAN Notices**, Volume 38 Issue 6

Full text available:  pdf(342.35 KB) Additional Information: [full citation](#), [abstract](#), [references](#)

Abstract machines have been widely employed in computing systems in order to obtain different aims. Compiler simplification, platform neutrality, code distribution, interoperability, and direct support for specific paradigms are examples of the benefits they offer. Although performance has been its main drawback, the use of modern techniques like adaptive (hotspot) just in time compilation has overcome this weakness. Nowadays, well-known platforms based on abstract machines such as Java™ o ...

Keywords: abstract machine, adaptability, extensibility, heterogeneously, reflection, virtual machine

7 [The FINITE STRING Newsletter: Abstracts of current literature](#) 

Computational Linguistics Staff

January 1987 **Computational Linguistics**, Volume 13 Issue 1-2

Full text available:  pdf(6.15 MB)  Additional Information: [full citation](#), [Publisher Site](#)

8 [Object-oriented software for quadratic programming](#) 

E. Michael Gertz, Stephen J. Wright

March 2003 **ACM Transactions on Mathematical Software (TOMS)**, Volume 29 Issue 1

Full text available:  pdf(216.18 KB) Additional Information: [full citation](#), [abstract](#), [references](#), [index terms](#)

The object-oriented software package OOQP for solving convex quadratic programming problems (QP) is described. The primal-dual interior point algorithms supplied by OOQP are implemented in a way that is largely independent of the problem structure. Users may

exploit problem structure by supplying linear algebra, problem data, and variable classes that are customized to their particular applications. The OOQP distribution contains default implementations that solve several important QP problem ty ...

Keywords: Interior-Point Methods, Object-Oriented Software, Quadratic Programming

- 9 [Persistent memory: a storage architecture for object-oriented database systems](#) 
Satish M. Thatte
September 1986 **Proceedings on the 1986 international workshop on Object-oriented database systems**

Full text available:  pdf(1.13 MB)

Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#)

Object-oriented databases are needed to support database objects with a wide variety of types and structures. A persistent memory system provides a storage architecture for long-term, reliable retention of objects with rich types and structures in the virtual memory itself. It is based on a uniform memory abstraction, which eliminates the distinction between transient objects (data structures) and persistent objects (files and databases), and therefore, allows the same set of powerful and f ...

- 10 [A survey of structured and object-oriented software specification methods and techniques](#) 
Roel Wieringa

December 1998 **ACM Computing Surveys (CSUR)**, Volume 30 Issue 4

Full text available:  pdf(605.26 KB)

Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#), [review](#)

This article surveys techniques used in structured and object-oriented software specification methods. The techniques are classified as techniques for the specification of external interaction and internal decomposition. The external specification techniques are further subdivided into techniques for the specification of functions, behavior, and communication. After surveying the techniques, we summarize the way they are used in structured and object-oriented methods and indicate ways in w ...

Keywords: languages

- 11 [ADAM: a language-independent, object-oriented, design environment for modeling inheritance and relationship variants in Ada 95, C++, and Eiffel](#) 
D. Needham, S. Demurjian, K. El Guemhioui, T. Peters, P. Zamani, M. McMahon, H. Ellis
December 1996 **Proceedings of the conference on TRI-Ada '96: disciplined software development with Ada**

Full text available:  pdf(1.45 MB)

Additional Information: [full citation](#), [references](#), [citations](#), [index terms](#)

- 12 [Using object-oriented thinking to teach Ada](#) 
John Ammirati, Mark Gerhardt, Dick Dye

July 1990 **Proceedings of the seventh Washington Ada symposium on Ada**

Full text available:  pdf(1.29 MB)

Additional Information: [full citation](#), [references](#), [citations](#), [index terms](#)

- 13 [An object-oriented language for symbolic computation—applied to machine element analysis](#) 

Lars Wiklund, Peter Fritzson

August 1992 Papers from the international symposium on Symbolic and algebraic computation

Full text available:  pdf(965.47 KB) Additional Information: [full citation](#), [references](#), [index terms](#)



14 Models and languages for parallel computation

David B. Skillicorn, Domenico Talia

June 1998 **ACM Computing Surveys (CSUR)**, Volume 30 Issue 2

Full text available:  pdf(298.05 KB) Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#)

We survey parallel programming models and languages using six criteria to assess their suitability for realistic portable parallel programming. We argue that an ideal model should be easy to program, should have a software development methodology, should be architecture-independent, should be easy to understand, should guarantee performance, and should provide accurate information about the cost of programs. These criteria reflect our belief that developments in parallelism must be driven by ...

Keywords: general-purpose parallel computation, logic programming languages, object-oriented languages, parallel programming languages, parallel programming models, software development methods, taxonomy



15 Object-oriented parallel computation for plasma simulation

Charles D. Norton, Boleslaw K. Szymanski, Viktor K. Decyk

October 1995 **Communications of the ACM**, Volume 38 Issue 10

Full text available:  pdf(421.61 KB) Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#)

Object-oriented techniques promise to improve the software design and programming process by providing an application-oriented view of programming while facilitating modification and reuse. Since the software design crisis is particularly acute in parallel computation, these techniques have stirred the interest of the scientific parallel computing community. Large-scale applications of ever-growing complexity, particularly in the physical sciences and engineering, require parallel processing ...



16 Type theories and object-oriented programming

Scott Danforth, Chris Tomlinson

March 1988 **ACM Computing Surveys (CSUR)**, Volume 20 Issue 1

Full text available:  pdf(4.39 MB) Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#)

Object-oriented programming is becoming a popular approach to the construction of complex software systems. Benefits of object orientation include support for modular design, code sharing, and extensibility. In order to make the most of these advantages, a type theory for objects and their interactions should be developed to aid checking and controlled derivation of programs and to support early binding of code bodies for efficiency. As a step in this direction, this paper surveys a number ...



17 Fast detection of communication patterns in distributed executions

Thomas Kunz, Michiel F. H. Seuren

November 1997 **Proceedings of the 1997 conference of the Centre for Advanced Studies on Collaborative research**

Full text available:  pdf(4.21 MB) Additional Information: [full citation](#), [abstract](#), [references](#), [index terms](#)

Understanding distributed applications is a tedious and difficult task. Visualizations based on process-time diagrams are often used to obtain a better understanding of the execution of the application. The visualization tool we use is Poet, an event tracer developed at the University of Waterloo. However, these diagrams are often very complex and do not provide the user with the desired overview of the application. In our experience, such tools display repeated occurrences of non-trivial commun ...

18 Special issue on knowledge representation

Ronald J. Brachman, Brian C. Smith
February 1980 **ACM SIGART Bulletin**, Issue 70

Full text available:  pdf(13.13 MB) Additional Information: [full citation](#), [abstract](#)

In the fall of 1978 we decided to produce a special issue of the SIGART Newsletter devoted to a survey of current knowledge representation research. We felt that there were two useful functions such an issue could serve. First, we hoped to elicit a clear picture of how people working in this subdiscipline understand knowledge representation research, to illuminate the issues on which current research is focused, and to catalogue what approaches and techniques are currently being developed. Second ...

19 AVANCE: an object management system

Anders Björnerstedt, Stefan Britts
January 1988 **ACM SIGPLAN Notices, Conference proceedings on Object-oriented programming systems, languages and applications**, Volume 23 Issue 11

Full text available:  pdf(1.87 MB) Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#)

AVANCE1 is an integrated application development and run-time system. It provides facilities for programming with shared and persistent objects, transactions and processes. The architecture is designed with decentralization in mind by having a large object identifier space and a remote procedure call interface to objects. Emphasis in this paper is on the programming language PAL and its relation with the underlying virtual machine.

20 Computing curricula 2001

September 2001 **Journal on Educational Resources in Computing (JERIC)**

Full text available:  pdf(613.63 KB)  html(2.78 KB) Additional Information: [full citation](#), [references](#), [citations](#), [index terms](#)

Results 1 - 20 of 200

Result page: [1](#) [2](#) [3](#) [4](#) [5](#) [6](#) [7](#) [8](#) [9](#) [10](#) [next](#)

The ACM Portal is published by the Association for Computing Machinery. Copyright © 2005 ACM, Inc.

[Terms of Usage](#) [Privacy Policy](#) [Code of Ethics](#) [Contact Us](#)

Useful downloads:  Adobe Acrobat  QuickTime  Windows Media Player  Real Player